A Java Toolbox for Analysis of MassIve Data STreams using Probabilistic Graphical Models

Masegosa, Andres; Martinez, Ana Maria; Ramos-López, Darío ; Langseth, Helge; Nielsen, Thomas Dyhre; Salmerón, Antonio; Cabanas, Rafael; Madsen, Anders Læsø

Publication date:
2016

A Java Toolbox for Analysis of MassIve Data STreams using Probabilistic Graphical Models

Andrés R. Masegosa[1], Ana M. Martínez[2], Darío Ramos-Lopez[3], Helge Langseth[1], Thomas D. Nielsen[2], Antonio Salmerón[3], Rafael Cabañas[2] & Anders L. Madsen[2,4]

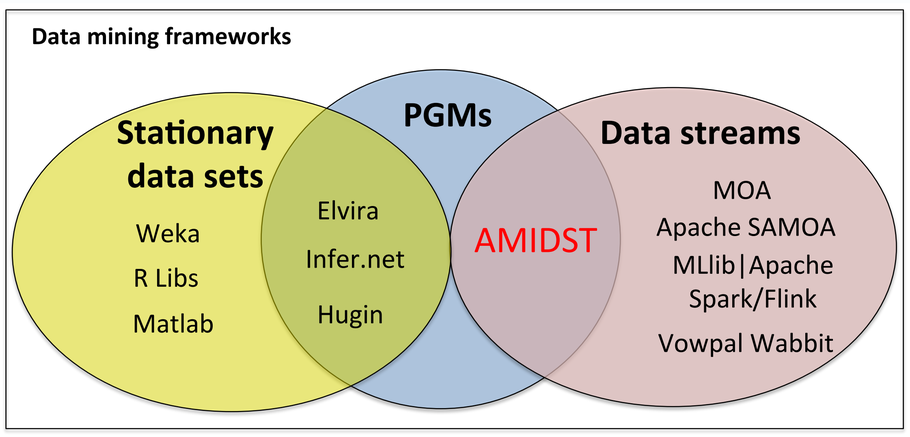
[1] Department of Computer and Information Science, NTNU, Norway

[2] Department of Computer Science, Aalborg University, Denmark

[3] Department of Mathematics, University of Almería, Spain

[4] Hugin Expert A/S, Aalborg, Denmark

Presentation



Academic and Industrial partners

Description

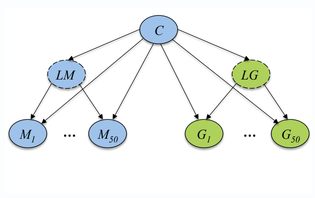
- **Analysis of big data streams**: A complete collection of algorithms for inference and learning of both static and dynamic Bayesian networks from streaming data. Existing software systems for PGMs only focus on stationary datasets.
- **Distributed parallel algorithms**: AMIDST provides parallel multi-core and distributed implementations of Bayesian parameter learning, using streaming variational Bayes and variational message passing.

	STATIC	DYNAMIC
Maximum Likelihood	• Naïve Bayes • TAN • (G)AODE/HODE	• Dynamic NB
Bayesian learning	• Gaussian Discriminant Analysis • Latent Classification Models (LCM) • Gaussian Mixtures • Bayesian Linear Regression • Factor Analysis • Mixture of FA	• Dynamic LCM • Hidden Markov Model (HMM) • Kalman Filter (KF) • Switching KF • Factorial HMM • Auto-regressive HMM • Input-Output HMM

Main Features

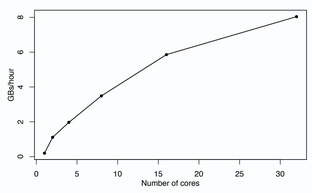
Java 8 based

Latent variable models

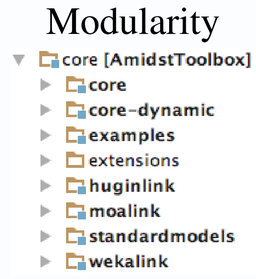


Integration

Big Data



Modularity



Open source

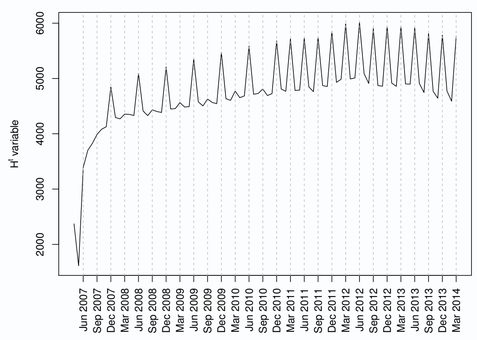
Code example

Learn hidden naive Bayes model from data stream

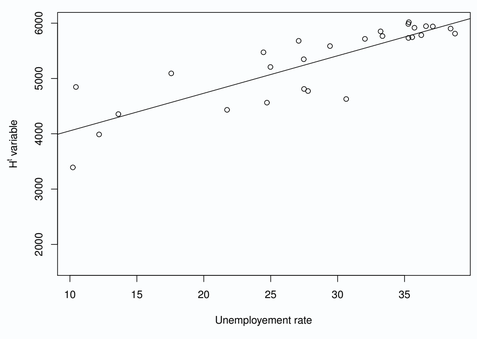
```
        //We can open the data stream using the static class DataStreamLoader
DataStream<DataInstance> data = DataStreamLoader.openFromFile("datasets/simulated/
  WasteIncineratorSample.arff");

//We create a SVB object
SVB parameterLearningAlgorithm = new SVB();

//We fix the DAG structure
parameterLearningAlgorithm.setDAG(DAGGenerator.getHiddenNaiveBayesStructure(data.
  getAttributes(),"GlobalHidden", 2));

//We fix the size of the window
parameterLearningAlgorithm.setWindowsSize(100);

//We can activate the output
parameterLearningAlgorithm.setOutput(true);

//We set the data which is going to be used for leaning the parameters
parameterLearningAlgorithm.setDataStream(data);

//We perform the learning
parameterLearningAlgorithm.runLearning();

//And we get the model
BayesianNetwork bnModel = parameterLearningAlgorithm.getLearntBayesianNetwork();

//We print the model
System.out.println(bnModel.toString());
```

Use-case: Risk prediction in credit operations

Concept drift



Correlated with Unemployment Rate



And much more...

amidst.eu

amidst.github.io/toolbox/

AMIDST project has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 619209.